A Federated Learning Approach for Link Prediction in Social Learning Networks with Varying Link Definitions

Jeremy Wong, University of California, San Diego

Computer Engineering master's student at University of California San Diego with experience in machine learning and neural networks. Working as a research assistant under Dr. Rajeev Sahay to understand how machine learning can be used to improve academic outcomes.

Dr. Rajeev Sahay, University of California, San Diego

Rajeev Sahay received the B.S. degree in electrical engineering from The University of Utah, Salt Lake City, UT, USA, in 2018, and the M.S. and Ph.D. degrees in electrical and computer engineering from Purdue University, West Lafayette, IN, USA, in 2021 and 2022, respectively. Currently, he is a faculty member in the Department of Electrical and Computer Engineering at UC San Diego. In 2021, he was the recipient of the Purdue Engineering Dean's Teaching Fellowship. Dr. Sahay's teaching interests include introductory and advanced programming (in C, C++, and Python) in the electrical engineering curriculum. His research interests lie in the intersection of networking and machine learning, especially in their applications to wireless communications and engineering education.

A Federated Learning Approach for Link Prediction in Social Learning Networks with Varying Link Definitions

Abstract

Collaborative learning improves student outcomes, but analyzing collaboration in in-person classrooms is difficult due to challenges in capturing interactions and differences from online settings, where interactions are digitally tracked. Social learning networks (SLNs) - computational graphs, where nodes represent students and links or edges between nodes represent student interactions - have been effective for modeling student interactions in online courses, yet their application to in-person classrooms and across diverse courses remains limited and inconsistent. Additionally, privacy regulations such as the Family Educational Rights and Privacy Act (FERPA) prevent centralizing student interaction data across institutions. To address these challenges, this work proposes a federated link prediction framework using feature-based feedforward neural networks trained via the Federated Averaging algorithm, enabling learning across heterogeneous courses without sharing raw data. The framework is modality-agnostic, supporting both online and in-person classrooms, where student interactions and in turn link definitions in SLNs may vary, while preserving student privacy. Experiments across multiple MOOC and in-person datasets show improved link prediction performance over baseline methods, particularly in sparse or low-engagement settings, demonstrating the framework's potential to support effective collaboration and group formation in diverse educational environments.

1 Introduction

Many educators and researchers have sought to understand how to improve student academic outcomes, both to enhance student learning and the quality of instruction. Several studies have considered collaborative learning, in which peers work together to complete academic activities, as a method of strengthening learning [1] [2]. This method includes working on assignments or projects, taking group assessments, or generally participating in group discussions. Collaborative learning has been shown to cultivate environments that allow students to model and practice cooperation, promote critical thinking skills, and develop positive attitudes of students toward teachers [1]. Other benefits of this strategy include higher retention rates, better academic outcomes, and greater motivation among students [2]. However, student groupings for collaborative learning activities are typically performed randomly either by-will of instructors or by-will of students. As a result, collaborative learning activities do not always maximize student learning. Thus, a critical challenge of collaborative learning is identifying effective student pairings that are mutually beneficial.

One method that has emerged to effectively model student collaborations in a course is a *Social Learning Network* (SLN), which is a computational graph in which students are represented by nodes, and edges, or links, represent interactions or group formations between the students. SLNs are dynamic models that evolve over the duration of a course. The graph begins as a set of nodes at the beginning of the course with no links, indicating no prior student interactions. As the course progresses and students begin interacting, on their own or via structured collaborative learning activities, the SLN evolves by forming edges between students that have interacted. Moreover, the dynamic nature of SLNs allows instructors to capture behavioral characteristics, such as the effectiveness of student interaction, within the model as well. Using the state of an SLN at an earlier point in the course (e.g., after half of the course has progressed), instructors can use SLNs to perform *link prediction*, which is the prediction of which edges in the SLN are likely to form between nodes (i.e., which students are likely to interact), at some point in the future (e.g., for a final project towards the end of the course) [3]. Such link prediction methods are known to optimize student groupings and enhance collaborative learning activities [4] as they are based on prior effective and ineffective pairings in the SLN.

Although link prediction can be performed in isolation on each classroom using machine learning (ML) approaches such as deep learning (DL) [5], recent studies have shown that combining SLN data from multiple classrooms can increase prediction performance and help with the cold start problem (i.e., predicting links early in the course when little to no links have formed) [6]. However, not all courses define links in the same way, with some using direct interactions as links, some using replies on discussion forums, and some using group or pair formations. Not only do these different data types pose a challenge for analysis across various courses, but different course modalities (i.e., in-person vs. online) also present an obstacle to creating a unified link prediction model due to the differences of student behavior in a physical classroom from student behavior in an online one. Even with a model that can synthesize data from various sources, which may produce an overall better model due to the diversity of the data, the issue of data privacy arises. Specifically, the Family Education Rights and Privacy Act (FERPA) restricts the sharing of student education records between institutions. Therefore, the centralization of student data from multiple courses or institutions into one data store is prohibited, rendering a unified model that predicts on data from multiple courses unusable.

Recently, federated learning (FL) has emerged as a promising privacy-preserving ML approach, which operates on independent data silos (also referred to as clients) without requiring the central aggregation of data. Instead, FL trains models on local data (e.g., SLN data hosted by a particular institution) and the model, rather than the data itself, is periodically aggregated into a global model, aiming to combine learned characteristics from each local client without transferring data away from any particular server or client.

In this work, we develop an FL-based framework for performing link prediction on courses with different behavioral data and modalities while preserving data privacy between them. By utilizing a greater quantity and diversity of course data, our framework improves its link prediction performance across a variety of courses while adhering to strict privacy standards by design of the FL paradigm. We apply our framework to courses of different sizes, modalities, and link definitions to produce a robust model that can perform effective link prediction in a diverse range of academic settings. The SLNs for each of these courses are constructed purely from student interaction data to enable the prediction of future connections between students based not on data about each individual student such as academic or demographic information, but on how students behave with one another in different environments. This work thus provides a data-driven framework for educators and instructors to optimize student connections to improve collaborative learning experiences across courses in a department or university regardless of course modality, size, or link definition in SLNs in a privacy preserving manner.

Overall, we make the following contributions in this work:

- To our best knowledge, we implement the first link prediction framework capable of operating on courses with different modalities and link definitions.
- We adhere to FERPA data privacy policies by preventing the sharing of data between different courses or the consolidation of data into one central data location.
- We demonstrate the feasibility of our method using data from different types of courses, including online and in-person courses, each with distinct link definitions in their SLNs.

2 Related Work

This section examines current research related to our proposed framework. Sec. 2.1 begins with a review of SLN formation. Sec. 2.2 subsequently discusses ML approaches for performing link prediction in a variety of computational graphs (including but not limited to SLNs), and lastly, Sec. 2.3 discusses existing work on FL on graph data, including but not limited to SLNs.

2.1 Social Learning Networks and Link Prediction

Social learning networks can be used to model any graph of students interconnected by interactions within an academic context, and these graphs can be analyzed using link prediction models. However, the current literature focuses primarily on online academic environments due to the abundance of automatically recorded data and the accessibility of virtual courses [4]. Several works investigate the patterns and behaviors present in collaborative online learning environments, such as Massive Open Online Courses (MOOCs) or LMS discussion forums [7] [8] [9]. They employ graph theory, natural language processing, and content analysis to form SLNs using student interactions from discussion forums and live classroom chats. While such analyses produce results that corroborate research in collaborative learning—that student academic performance improves with greater engagement with peers—these works do not provide a means of recommending student connections to enhance such collaborative learning experiences. Others use social media sites or blogs as a medium for SLNs [10] [11], and some researchers even form groups through social network analysis (SNA) on these sites [12]. However, such SNA frameworks are built specifically for blogs and social media sites and are applied to a single course, limiting its reuse to a specific subset of SLNs. Despite this shortcoming, the results of the SNA group formation demonstrate the efficacy of this method in improving student learning outcomes, motivating the use of link prediction to produce meaningful groups of students. Additional research focuses on using node information, such as learning styles in an academic setting or number of followers in a social media setting, to similarly recommend connections between individuals [3] [13] [14]. Other studies analyze social interaction information, including likes, comments, or replies, to form predictions [15] [16] [17] [18]. Although these studies produce promising results for link prediction in SLNs and social networks in general, they each face the aforementioned limitation of specializing in a single data format, preventing the synthesis of data across various courses. They also employ simpler recommendation frameworks, which may not suit more complex network structures such as jointly analyzing behavior across classrooms.

2.2 Using Neural Networks for Link Prediction

Link prediction is often performed using simpler algorithms or ML models, but neural networks have received more recent attention due to their ability to perform well on complex networks. Whether for protein structures, Internet of Things device connections, or social networks, various neural network architectures can be used to predict future link existence or values [19] [20] [21] [22] [23]. These studies use link values and features to infer link formation, which is helpful when considering link formation in an SLN since the data available in an SLN often is link data [17]. However, though these works cover both larger and smaller datasets individually, prior work paper does not synthesize datasets of the same order of magnitude as typical course datasets using one cohesive model. Thus, they are less suitable for making predictions in a variety of courses. Certain works partially address this shortcoming by evaluating multiple datasets with mixed sizes and data types more similar to online or in-person courses, demonstrating success across the tested datasets [24] [25]. These models exhibit their efficacy when training isolated models on each dataset, but are incapable of utilizing multiple data sources to train one central model. The link predictions are thus limited to each model and cannot be improved with data external to that model's data source. To help synthesize multiple data sources into one model, some propose subgraph-based and decentralized learning approaches for neural networks in link prediction to leverage commonalities among node neighborhoods and to distinguish between nodes whose neighbors are not constantly available [26] [22] [27] [28]. While these methods can be likened to a model using isolated SLNs to train and infer on, the networks are still connected to some degree, which, in the context of a university course, would be impermissible, as data cannot be shared among courses. The SLNs must maintain strict separation from one another, but these works serve different purposes and consequently do not enforce this standard.

2.3 Federated Learning on Graphs

To address privacy concerns about sharing data from or between different sources, federated learning (FL) has been proposed as an alternative to traditional centralized learning. FL is a decentralized approach to learning in which, instead of sharing data between multiple sources, or clients, model parameters are shared to a central location, aggregated, and then redistributed to clients to preserve privacy and reduce computational overhead [29]. Federated models are typically used to train ML models when the training data is distributed over multiple clients and centralizing the data is not feasible due to size or privacy concerns, including in social recommendation, wireless network optimization, and data recovery [30–34]. These works reveal the potential of federated models when used on graph data, and successfully enforce data privacy between clients, [30] being especially useful, as SLNs are analyzed similarly to social

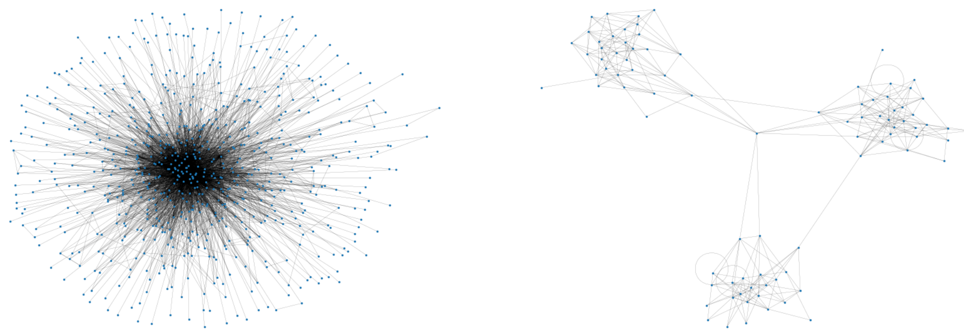

Figure 1: Visualization of MOOC (left) and in-person (right) SLNs.

networks [6]. However, these works use datasets that are much larger than in-person or online course datasets. [34] also performs node classification, which, while useful in certain applications, does not help with student connection recommendation. Another study uses a smaller network structure to promote the use of FL models in graphs with fewer nodes [35], where the previous issue of having very large graphs through its 33-node structure is addressed, but also introduces the opposite problem of only training on small networks. With a combination of larger and smaller datasets, the proportion of sampled data will vary greatly, which can cause performance issues for models that train on more data. [36] provides a balance between larger and smaller datasets, as it is designed for SLN analysis on MOOC course data. Although these prior works provide a framework for performing link prediction on SLNs in a federated environment, they do not jointly consider in-person and online courses, where link definitions and, in turn, student behavior towards collaborative learning, can greatly differ.

3 Methodology

This section details the methodology for our proposed framework. Sec. 3.1 begins by describing our representation of SLNs. Sec. 3.2 then discusses the formation of links in the different types of courses that we consider. Sec. 3.3 describes our feature engineering approach. Sec. 3.4 defines our federated link prediction model. Lastly, Sec. 3.5 explains the training setup and evaluation metrics for our framework.

3.1 SLN Construction

To mathematically represent an SLN, let us define the number of nodes in each graph $\mathcal{G}$ (i.e., an SLN) as N, where N is the number of students enrolled in the course. The set of all students $\mathcal{S}$ in the course can be written as $\{S_1, S_2, ..., S_N\}$. For each link defined as (u, v), S_u represents the source student node, and S_v represents the target student node. We will further elaborate on how links are defined in different SLNs in Sec. 3.2. We use an unweighted graph, which means that for each link weight $w_{u,v}$ that corresponds to the link (u, v), $w_{u,v} \in \{0, 1\}$. For example, if there exists a link from student S_1 to student S_2, then $w_{1,2} = 1$. Conversely, if there exists no link from student S_2 to student S_1, then $w_{2,1} = 0$. Figure 1 visualizes two sample SLNs constructed from our data, which continue to form throughout the course.

3.2 Link Formation

We consider three distinct approaches for collecting SLN data and defining links. The first approach uses data from MOOC discussion forums, where an interaction between two students on the forum is considered a link and assigned a weight of 1. Any other student pairing where there was no interaction is considered a non-link, or a link with a weight of 0. The second approach uses in-person interaction data created using surveys that were submitted after completing weekly collaborative learning assignments. Each student was assigned a new random partner each week with whom to complete the assignment. The surveys asked students to rate the quality of their experiences on a scale from 1 to 10 using the following questions:

- *To what extent did the interaction help you understand the course material better?*
- *How engaged did you feel during the interaction?*
- *How equally do you think you both contributed to the discussion?*
- *How well were you able to communicate your ideas during the interaction?*
- *How would you rate the flow of the conversation?*
- *How would you rate your overall satisfaction in working with your assigned partner today?*

We binarized this dataset by setting the weight to 0 for ratings from 1 to 5, and 1 for ratings from 6 to 10. Pairs of students who did not interact were assigned a weight of 0.

Each of these datasets is composed of all the student interactions that occur throughout the duration of the respective course. The students were assigned a unique identifier that replaced all personally identifiable information, and these identifiers were recorded in each dataset. Using this data, we can form the SLN graphs described in Sec. 3.1. Our framework is compatible with both of the link definitions considered above. With the SLNs created from the data, we can perform feature engineering on the data to make it compatible with our feature-based feedforward model.

3.3 Graph Feature Engineering

To produce inputs that our feature-based model can use (our feature-based model is further described in Sec. 3.4), we convert the adjacency matrices of the SLNs, which are square matrices that represent links between nodes, to feature matrices based on computed graph features. Using $\Gamma(u)$ as the set of neighbors of node u and $k(u)$ as the degree of node u, we compute the following six features to construct our feature matrices:

1. *Jaccard Coefficient* [37]: This coefficient measures similarity based on the ratio of the number of shared features to the total number of features. For a selected graph $\mathcal{G}$, these features represent the students with whom student u has interacted. The Jaccard coefficient is defined by the following:

$$Jaccard(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|} \quad (1)$$

2. *Adamic-Adar (AA) Index* [37] [38]: This index is determined by the shared neighbors among students. The AA index uses the logarithmic function to assign more weight to students with fewer connections than students with more connections. The Adamic-Adar index is expressed as follows:

$$AA(u, v) = \sum_{(i \in \lceil(u) \cap (v)\rceil)} \frac{1}{\log \lceil(i)\rceil} \tag{2}$$

3. *Resource Allocation (RA) Index* [37] [38]: This metric allows the sharing of resources between unconnected nodes u and v via a common node. This common node allocates a fraction of its neighbors' resources to each other, where this allocation is inversely proportional to the node degrees, reducing the effect of higher-order nodes. The following equation defines the resource allocation index:

$$RA(u, v) = \sum\nolimits_{z \in \lceil(u) \cap \lceil(v)} \frac{1}{k(z)} \tag{3}$$

4. *Preferential Attachment (PA) Score* [37] [38]: This score is based on the idea that highly-connected nodes have a higher probability of forming more connections than nodes with fewer connections. In the context of SLNs, PA measures the tendency of students with more interactions to interact with new students. The preferential attachment score is expressed by the following:

$$PA(u, v) = |\Gamma(u).\Gamma(v)| \tag{4}$$

5. *Cosine Similarity (CS) Index* [37] [38]: This index uses the cosine of the angle formed by the vectors of the neighborhoods of two nodes to determine similarity. A larger CS index implies the two nodes are more similar. The cosine similarity index is defined as follows:

$$CS(u, v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{\sqrt{k(u)k(v)}} \tag{5}$$

6. *Dice Coefficient* [39]: This coefficient shares similarities to the Jaccard coefficient in that they both depend on the intersection of two nodes' neighbors. The Dice coefficient, however, emphasizes the common student interactions between the two nodes more than the Jaccard coefficient does. The Dice similarity coefficient is calculated by the following:

$$Dice(u, v) = \frac{2 \cdot |\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u)| + |\Gamma(v)|} \tag{6}$$

Thus, the dimensions of each feature matrix is $6 \times N \times N$, where N represents the number of students in a given dataset. For each pair of students, we compute the $6 \times 1 \times 1$ feature vector using the features described above. These vectors serve as the link prediction model input.

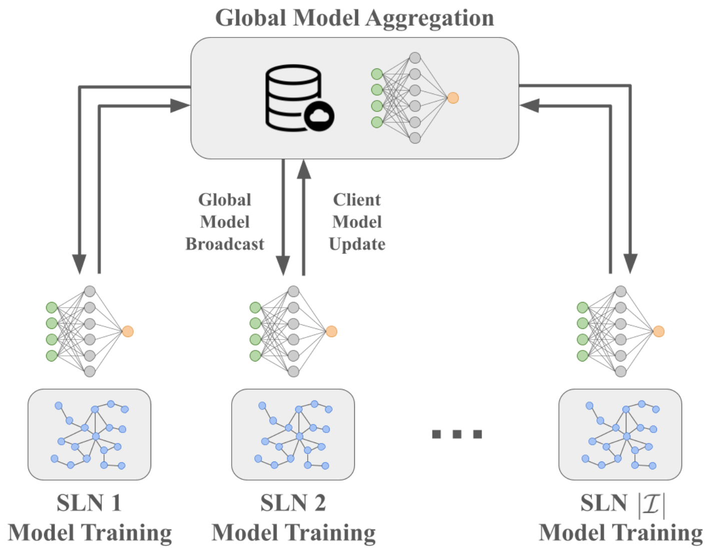

Figure 2: Visualization of the client model training and update, and global model aggregation and broadcasting in the federated learning algorithm [36].

3.4 Federated Learning for Link Prediction

Here, we use the feature-engineered data from the previous section to perform link prediction. The link prediction model that we select is a feedforward neural network (FNN), a fundamental DL architecture that moves data in only one direction: from the input through one or more hidden layers to the output. FNNs are simple, yet powerful, models that are primarily used in pattern recognition tasks such as image classification or speech recognition, but can also be applied to other problems like link prediction. To use a graph structure as input to an FNN, we create feature matrices from the adjacency matrices formed by the graph data as discussed in Sec. 3.4

FNNs, however, require data to be centralized to a single location for training, thus necessitating the aggregation of data to a single location from multiple classrooms or institutions to perform effective training. The rationale for training a model on data from multiple classrooms comes from the well-known concept that providing more data to an ML model helps the model generalize better due to the greater diversity of data being trained on. Thus, if our link prediction model is trained on data from multiple courses, the model would likely improve in performance. However, as mentioned before, FERPA prohibits sharing student data between courses. Thus, a single central model cannot train directly on data collected from multiple classrooms.

To address this challenge, we adopt a distributed training paradigm known as federated learning (FL). FL is a decentralized learning approach that trains models on local clients (i.e., classrooms) and aggregates the local model parameters of each client to a global model stored on a central server [29]. Local client data, which in our context is each course's student data, is never shared outside of the course. Rather, each course, denoted as i, has its own FNN model that trains solely on the course's data $\mathcal{D}_i$ over a certain number of epochs, which we will call e_{max} [36]. The model parameters are then sent to the global model for aggregation, which, for our framework, is the federated averaging (FedAvg) algorithm. FedAvg aggregates the local parameters by computing the weighted average of each client's local parameters. The global model then redistributes the global parameters back to the local models for further training and aggregation. This process repeats for $k_{\max}$ communication rounds. We visualize our federated learning link prediction algorithm in Figure 2.

Mathematically, we can define the FedAvg process as follows. For each epoch $e \in \{1, 2, ..., e_{max}\}$, we have this local update equation:

$$\mathbf{w}_i^{(k),e} = \mathbf{w}_i^{(k),e-1} - \eta\, \tilde{\nabla}\mathcal{L}_i\left(\mathbf{w}_i^{(k),e-1}\right), \tag{7}$$

where $\mathbf{w}_i^{(k),e}$ represents the local weights $\mathbf{w}$ for the i-th client on the k-th communication round and e-th local epoch. η is the local learning rate, and $\tilde{\nabla}\mathcal{L}_i\left(\mathbf{w}_i^{(k),e-1}\right)$ notates the stochastic approximate gradient of the loss function of each client model, defined by the following:

$$\tilde{\nabla}\mathcal{L}_i\left(\mathbf{w}_i^{(k),e-1}\right) = \frac{1}{\left|\mathcal{B}_i^{(k),e}\right|} \sum_{d \in \mathcal{B}_i^{(k),e}} \nabla\ell\left(\mathbf{w}_i^{(k),e-1}; d\right), \tag{8}$$

where $B_i^{(k),e} \subseteq \mathcal{D}_i$ is the random mini-batch of the data selected from a given course dataset, $|B_i^{(k),e}|$ is the mini-batch size, and $\ell\left(\mathbf{w}_i^{(k),e-1}; d\right)$ is the loss function of the local model for weight $\mathbf{w}_i^{(k),e-1}$ and data point d. The course clients then send their local parameters $\mathbf{w}_i^{(k),e_{\max}}$ to the server aggregation, which is notated by the below equation:

$$\mathbf{w}^{(k+1)} = \sum_{i \in \mathcal{I}} \frac{|\mathcal{D}_i|\mathbf{w}_i^{(k),e_{\max}}}{|\mathcal{D}|}, \tag{9}$$

where $|\mathcal{D}_i|$ is the size of dataset $\mathcal{D}_i$, and $|\mathcal{D}| = \sum_{i \in \mathcal{I}} |\mathcal{D}_i|$ represents the total size of all the datasets of the courses. The model parameters of each local client $\mathbf{w}_i^{(k+1)}$ are then updated with the global model parameters $\mathbf{w}^{(k+1)}$, and the federated training process repeats.

This method addresses two key issues with traditional, or centralized, learning. Namely, FL prevents (i) privacy breaches between clients or from clients to the server and (ii) increased computational load from data transfers and training on large amounts of data. Using FL, we train a global link prediction model without having to access the raw course data, allowing us to comply with FERPA while enhancing the generalization capabilities of our model across courses with varying modalities, enrollment sizes, and link definitions.

3.5 Training and Evaluation

We now describe the training and evaluation of our federated link prediction framework. We train our local FNN models using the Adam optimizer and use binary-cross-entropy-with-logits loss for our loss function. We evaluate the performance of our global model using accuracy and area-under-the-curve (AUC), which are standard evaluation metrics in ML.

Table 1: Course and SLN information for MOOC [36], non-MOOC, and in-person datasets.

Course Title	Abbreviation	Nodes	Links	Course Type
Algorithms Design and Analysis I	algo	1165	6773	MOOC
English Composition I	comp	900	3418	MOOC
Machine Learning	ml	3290	30610	MOOC
Shakespeare in Community	shak	677	4702	MOOC
Intro to Computer Engineering	fa24	87	295	In-Person

Accuracy is the ratio of the number of correct predictions to the total number of predictions:

$$\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{10}$$

where TP, TN, FP, and FN are the numbers of true positives, true negatives, false positives, and false negatives, respectively.

AUC is the area under the receiver operating characteristic (ROC) curve, which visualizes the binary classification performance of a model. The curve is plotted with true positive rate (TPR) on the vertical axis and false positive rate (FPR) on the horizontal axis, where TPR and FPR are defined by the following equations:

$$TPR = \frac{TP}{TP + FN} \qquad FPR = \frac{FP}{TN + FP} \tag{11}$$

AUC values fall within the range of $0 < AUC < 1$. An AUC of 0.5 means that a model achieves a classification performance equal to that of random selection, while an AUC of 1 means that a model perfectly distinguishes between classes. For this work, AUC is more useful than accuracy because of the large class imbalances in the SLNs due to their inherent sparsity as the number of unformed links between student pairs greatly exceeds the number of formed links. Thus, while accuracy may be artificially inflated as a consequence of the class imbalance of SLNs, AUC will not, which is why we will consider AUC more heavily for evaluation.

4 Results and Discussion

In this section, we perform an empirical evaluation of our proposed framework. Sec. 4.1 describes the datasets used in our experiments and our model setup. Next, Sec. 4.2 details the baseline models with which we compare our proposed model. Lastly, Sec. 4.3 reports the experimental results of the different models, FL networks, and considered courses.

4.1 Empirical Setup

As shown in Table 1, we use four discussion forum-based MOOC datasets and one in-person classroom behavioral-based dataset. The courses from which the datasets are collected cover a wide variety of topics: *Algorithms Design and Analysis I* (algo), *English Composition I* (comp), *Machine Learning* (ml), *Shakespeare in Community* (shak), and *Introduction to Computer*

Engineering (Fall 2024) (fa24). In terms of size, these datasets vary greatly as well, especially between the MOOC datasets and in-person dataset, capturing high diversity among the considered classrooms. The MOOC datasets, as discussed in Sec. 3.2, are drawn from logs of student interactions on the MOOC forums. These interactions were assigned with either a link being formed (i.e., a pair of students interacting on a forum), notated as a 1, or a link not being formed (i.e., a pair of students never interacting on a forum), notated as a 0.

The most unique of these datasets is the *Introduction to Computer Engineering* dataset due to the method by which it was gathered. Unlike the other four datasets, this data was obtained through the administration of Google Form surveys (described in Sec. 3.2) at the end of weekly in-person lab sections. The course was an undergraduate electrical engineering course at the University of California San Diego and lasted ten weeks, with nine weeks of data. Each lab section, students would participate in a collaborative learning exercise, where they were randomly partnered with another student. When the students completed the assignment or the lab section was finished, students completed the survey to evaluate their collaborative learning experience, providing a quantified measure of the quality of their in-person interaction.

The FNN model we implement uses one input, two hidden, and one output layer, with 64 and 32 units in the first and second hidden layer, respectively. We also perform batch normalization and use the Leaky ReLU activation function with a negative slope of 0.01 after the first and second layers, and dropout with $p = 0.2$ after the first layer. We set the learning rate of our Adam optimizer to 0.0005 and our batch size to 32. We train our client models each for 2 epochs before aggregating the local parameters and repeat the federated communication cycle for 30 rounds. These hyperparameters were selected by performing a grid search and selecting the best performing parameters. Each dataset undergoes an 80-20 train-test split, where the same data is used across all models for each client. We also confirm that the batches use the same data for each model over each local epoch and global communication round to ensure that the models are being compared fairly.

4.2 Baseline Models

We compare our proposed framework with three baseline models, which are described here.

Convolutional Neural Network (CNN): This DL architecture relies on convolutional layers that use kernels to manipulate the dimensionality of the inputs to these layers [40]. CNNs are especially useful for image recognition tasks due to their focus on spatial dimensionality and learning features within images. Feature matrices derived from the SLN graph, which will be elaborated on, act as an image-like structure and can be used as input to the CNN. The CNN we implement uses two convolutional layers with ReLU activations after each layer.

Graph Neural Network (GNN): This DL architecture trains on nodes and edges and forms and infers embeddings within the model. Specifically, we implement a GraphSAGE model which, in the context of an SLN, creates node embeddings from student information hidden within the model [41]. With this algorithm, student nodes can distribute information and learn from neighboring nodes so that each node becomes more aware of its neighborhood and the complete graph structure. The GNN, having learned about the overall structure, can then infer future link formation. Thus, the GNN does not use the engineered features used for our feature-based FNN

Table 2: Final AUC and ACC values for an FL network consisting of MOOC datasets, showing that our proposed feature-based framework is the top-performing model in terms of AUC and ACC, while the AUCs and accuracies of the GNN are significantly lower than the other models. The model with the best performance for each dataset and metric is shown in bold.

Dataset	Metric	FNN	CNN	GNN	LR
algo	AUC	**0.9287**	0.9284	0.8036	0.9285
	ACC	**0.8446**	**0.8446**	0.5952	**0.8446**
comp	AUC	**0.9465**	0.9455	0.7915	0.9457
	ACC	**0.8889**	**0.8889**	0.5943	**0.8889**
ml	AUC	**0.9298**	0.9277	0.6860	0.9289
	ACC	**0.8608**	**0.8608**	0.5602	**0.8608**
shak	AUC	**0.9114**	0.9086	0.7239	0.9098
	ACC	**0.8750**	**0.8750**	0.5755	**0.8750**

Table 3: Final AUC and ACC values for an FL network consisting of MOOC datasets and the in-person dataset, showing that, consistent with Table 2, the proposed feature-based framework is consistently the best-performing model in terms of AUC and ACC, while the non-feature based GNN is consistently the lowest performing.

Dataset	Metric	FNN	CNN	GNN	LR
algo	AUC	0.9257	0.9256	0.7874	**0.9258**
	ACC	**0.8391**	**0.8391**	0.5801	**0.8391**
comp	AUC	**0.9487**	0.9477	0.7901	0.9479
	ACC	**0.8925**	**0.8925**	0.5841	**0.8925**
ml	AUC	**0.9331**	0.9312	0.6887	0.9316
	ACC	**0.8598**	**0.8598**	0.5480	**0.8598**
shak	AUC	**0.9267**	0.9241	0.7401	0.9246
	ACC	**0.8894**	**0.8894**	0.5580	**0.8894**
fa24	AUC	**0.7122**	0.7085	0.6021	0.7107
	ACC	**0.6800**	**0.6800**	0.5450	**0.6800**

or baseline CNN. Our GNN uses two GraphSAGE convolutional layers with ReLU between them.

Logistic Regression (LR): This model performs classification by fitting a logistic function to the data. The model learns features and predicts based on the learned linear combination of weights that best fit the training data. The LR model uses the same input as the FNN.

4.3 Performance Analysis

We consider four different FL setups. The first includes only the MOOC datasets: *algo*, *comp*, *ml*, *shak*. The second uses the MOOC datasets with the in-person interaction dataset (*fa24*). The third trains on the three technical topics courses (*algo*, *ml*, *fa24*). The fourth consists of an FL network trained on the two humanities courses (*comp*, *shak*). These combinations allow us to ensure that our method works when the FL network consists of different types of SLNs.

Table 4: Final AUC and ACC values for an FL network consisting of technical MOOC datasets, showing that the proposed feature-based framework overall performed the best across both AUC and ACC, and where the framework did not, the values were almost identical to the best model.

Dataset	Metric	FNN	CNN	GNN	LR
algo	AUC	0.9416	0.9414	0.8143	**0.9417**
	ACC	**0.8450**	**0.8450**	0.6052	**0.8450**
ml	AUC	**0.9351**	0.9324	0.6865	0.9324
	ACC	**0.8584**	0.8576	0.5521	0.8576
fa24	AUC	0.7811	0.7811	0.5843	**0.7813**
	ACC	**0.7000**	**0.7000**	0.5300	**0.7000**

Table 5: Final AUC and ACC values for an FL network consisting of humanities MOOC datasets, showing that the proposed feature-based framework achieved the best results across both AUC and ACC for both *comp* and *shak* datasets.

Dataset	Metric	FNN	CNN	GNN	LR
comp	AUC	**0.9571**	0.9542	0.8153	0.9542
	ACC	**0.9006**	**0.9006**	0.6111	**0.9006**
shak	AUC	**0.9385**	0.9348	0.7621	0.9349
	ACC	**0.8936**	0.8931	0.5812	0.8931

From Tables 2, 3, 4, and 5, we observe that accuracy was identical or nearly identical across all three feature-based models and that the GNN accuracy was significantly lower than that of the other three models. Our FNN model performed especially well with the *comp* dataset, which may be due to the graph structure of this dataset in particular. We also note that each of the models struggled more with the in-person dataset, likely a result of the the dataset being significantly smaller than the others. However, as discussed previously, we consider AUC more strongly than accuracy due to the reasoning mentioned in Sec. 3.5. We see that the FNN has the best, if not close to the best, AUC in all or the majority of the datasets for each federated setup. Yet, we also observe that the federated FNN, CNN, and logistic regression models all had similar, if not the same, performances across each of the datasets. The shared feature-based nature of these models likely caused the similarities, which raises the question of whether a complex architecture, such as the CNN, is necessary for this application. Notably, however, we see that the GNN performs the worst in our experiments in terms of AUC, with significant differences between its performance and that of the feature-based methods. This result is consistent with the vast differences in dataset sizes as well as the GNN's ability to calculate its own features, which may not be ideal on small datasets.

5 Conclusion

In this work, we developed a privacy preserving federated learning link prediction framework for social learning networks capable of jointly operating in multiple classrooms with different link formation definitions. We implemented a feature-based model, specifically a feedforward neural network, to form predictions on future connections between students, and demonstrate the effectiveness of our framework using data from a variety of course modalities, topics, and sizes. Our model achieved particularly better results than state-of-the-art non-feature-based graph neural network, which are specifically designed to operate over graphs such as graph neural networks. Our proposed framework serves as a useful tool for instructors in generating beneficial groups for positive collaborative learning experiences. Future exploration of this work would benefit from training and evaluation with more in-person data to examine whether an improved balance of data in terms of course modality would produce better overall results. Future work will also consider comparing other feature-based models, or adding additional features, to determine if they can achieve similar results to the ones tested in this work.

Acknowledgments

This work was supported in part by the University of California San Diego Academic Senate under grant RG116457 and in part by the National Science Foundation under grant ECCS-2512912.

References

[1] M. Laal and S. M. Ghodsi, “Benefits of collaborative learning,” *Procedia - Social and Behavioral Sciences*, vol. 31, pp. 486–490, 2012, world Conference on Learning, Teaching & Administration - 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877042811030205

[2] T. Panitz and P. Panitz, *Encouraging the Use of Collaborative Learning in Higher Education*, 10 2018, pp. 161–202.

[3] B. Ramya, N. S. Varma, and R. Indra, “Recommendations in social network using link prediction technique,” in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, 2020, pp. 782–786.

[4] C. G. Brinton and M. Chiang, “Social learning networks: A brief survey,” in *2014 48th Annual Conference on Information Sciences and Systems (CISS)*, 2014, pp. 1–6.

[5] R. Sahay, S. Nicoll, M. Zhang, T.-Y. Yang, C. Joe-Wong, K. A. Douglas, and C. G. Brinton, “Predicting learning interactions in social learning networks: A deep learning enabled approach,” *IEEE/ACM Transactions on Networking*, vol. 31, no. 5, pp. 2086–2100, 2023.

[6] A. Hridi, M. Hoq, Z. Gao, C. Lynch, R. Sahay, S. Hosseinalipour, and B. Akram, “Privacy-preserving distributed link predictions among peers in online classrooms using federated learning,” in *Proceedings of the 18th International Conference on Educational Data Mining*, Palermo, Italy, July 2025, pp. 276–288.

[7] M. Saqr and C. S. Montero, “Learning and social networks - similarities, differences and impact,” in *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, 2020, pp. 135–139.

[8] Y. Zhen, J.-D. Luo, and H. Chen, "Prediction of academic performance of students in online live classroom interactions—an analysis using natural language processing and deep learning methods," *Journal of Social Computing*, vol. 4, no. 1, pp. 12–29, 2023.

[9] W. Zou, X. Hu, Z. Pan, C. Li, Y. Cai, and M. Liu, "Exploring the relationship between social presence and learners' prestige in mooc discussion forums using automated content analysis and social network analysis," *Computers in Human Behavior*, vol. 115, p. 106582, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0747563220303290

[10] A. Ban, N. C. Pa, J. Din, and N. A. Mohd Yaa'cob, "Integrating social collaborative features in learning management system: A case study," in *2017 IEEE Conference on e-Learning, e-Management and e-Services (IC3e)*, 2017, pp. 67–72.

[11] K. Halimi, H. Seridi, and C. Faron-Zucker, "Solearn: A social learning network," in *2011 International Conference on Computational Aspects of Social Networks (CASoN)*, 2011, pp. 130–135.

[12] S. Eneje, S. O. Sanni, and C. F. Pereira, "Engagement in a virtual learning on two social networks of an engineering course using the social network analysis- an approach using a case study," in *2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2020, pp. 1–6.

[13] L. Wang, "Social recommendation algorithm based on learning interests and styles for online blended learning platform," in *2022 3rd International Conference on Information Science and Education (ICISE-IE)*, 2022, pp. 199–204.

[14] G. K, R. Rajasekaran, V. V. Krishna Varun, D. Vuyyuru, and B. Thirumaleshwari Devi, "Link prediction in social networks using machine learning," in *2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, 2023, pp. 665–673.

[15] S. Souabi, A. Retbi, M. K. Idrissi, and S. Bennani, "A recommendation approach in social learning based on k-means clustering," in *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2020, pp. 1–5.

[16] ——, "A recommendation approach based on correlation and co-occurrence within social learning network," in *2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech)*, 2020, pp. 1–6.

[17] Y. Li, H. Bao, Y. Zheng, and Z. Huang, "Social analytics framework to boost recommendation in online learning communities," in *2015 IEEE 15th International Conference on Advanced Learning Technologies*, 2015, pp. 405–406.

[18] C.-M. Chen, C.-M. Hong, and C.-C. Chang, "Mining interactive social network for recommending appropriate learning partners in a web-based cooperative learning environment," in *2008 IEEE Conference on Cybernetics and Intelligent Systems*, 2008, pp. 642–647.

[19] M. Xu, J. Xu, Y. Xia, J. Xie, W. Liu, S. Hu, and D. Huang, "Recurrent neural network based link quality prediction for wireless sensor networks," in *2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS)*, 2021, pp. 905–909.

[20] X. Yang and Z. Zhang, "Spiking neural networks-based neighborhood co-occurrence link prediction," in *2024 4th International Conference on Computer Communication and Artificial Intelligence (CCAI)*, 2024, pp. 208–213.

[21] Z. Zhang, X. Wu, G. Zhu, W. Qin, and N. Liang, "A graph attention network-based link prediction method using link value estimation," *IEEE Access*, vol. 12, pp. 34–45, 2024.

[22] K. Zhu and M. Cao, "A semantic subgraphs based link prediction method for heterogeneous social networks with graph attention networks," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8.

[23] S. Yeom, C. Choi, S. S. Kolekar, and K. Kim, "Graph convolutional network based link state prediction," in *2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2021, pp. 246–249.

[24] M. G. Morshed, T. Sultana, and Y.-K. Lee, "Lel-gnn: Learnable edge sampling and line based graph neural network for link prediction," *IEEE Access*, vol. 11, pp. 56 083–56 097, 2023.

[25] M. Abedini and H. Shakibian, "Inter-layer similarity-based graph neural network for link prediction in social multiplex networks," in *2024 10th International Conference on Web Research (ICWR)*, 2024, pp. 76–80.

[26] K. Selvarajah, K. Ragunathan, Z. Kobti, and M. Kargar, "Dynamic network link prediction by learning effective subgraphs using cnn-lstm," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8.

[27] M. Li, Z. Wang, L. Liu, X. Liu, and W. Zhang, "Subgraph-aware graph kernel neural network for link prediction in biological networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 28, no. 7, pp. 4373–4381, 2024.

[28] E. Krasanakis, S. Papadopoulos, and I. Kompatsiaris, "p2pgnn: A decentralized graph neural network for node classification in peer-to-peer networks," *IEEE Access*, vol. 10, pp. 34 755–34 765, 2022.

[29] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," 2023. [Online]. Available: https://arxiv.org/abs/1602.05629

[30] Z. Xu, B. Li, and W. Cao, "Federated learning-based social recommendation with social denoising," in *2024 International Conference on New Trends in Computational Intelligence (NTCI)*, 2024, pp. 331–335.

[31] C. Ma, Y. Wang, and B. He, "Hierarchical social recommendation based on federated graph neural network," in *2024 International Conference on Artificial Intelligence and Digital Libraries (AIDL)*, 2024, pp. 79–84.

[32] Y. Zhou and Y. Li, "Federated meta-reinforcement learning for adaptive wireless networks: A unified framework integrating proximal policy optimization, meta-learning, and graph neural networks," in *2025 8th International Conference on Computer Information Science and Application Technology (CISAT)*, 2025, pp. 590–594.

[33] Z. Liu, Y. Jiang, J. Gu, D. Liu, Y. Song, A. Jiang, and P. Duan, "Federated graph neural networks based on multiscale residuals in industrial internet of things," *IEEE Transactions on Computational Social Systems*, vol. 12, no. 6, pp. 4999–5014, 2025.

[34] R. Lei, P. Wang, J. Zhao, L. Lan, J. Tao, C. Deng, J. Feng, X. Wang, and X. Guan, "Federated learning over coupled graphs," *IEEE Transactions on Parallel and Distributed Systems*, vol. 34, no. 4, pp. 1159–1172, 2023.

[35] H. Wu, Z. Xu, Y. Ding, W. Shi, A. Zhang, and Y. Liu, "Graph convolutional network-based federated learning method for distribution system topology identification," in *2024 China International Conference on Electricity Distribution (CICED)*, 2024, pp. 724–728.

[36] A. Hridi, M. Hoq, Z. Gao, C. Lynch, R. Sahay, S. Hosseinalipour, and B. Akram, "Privacy-preserving distributed link predictions among peers in online classrooms using federated learning," 2025. [Online]. Available: https://zenodo.org/doi/10.5281/zenodo.15870179

[37] A. JIBOUNI, D. LOTFI, M. E. MARRAKI, and A. HAMMOUCH, "A novel parameter free approach for link prediction," in *2018 6th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, 2018, pp. 1–6.

[38] S. U. Balvir, M. M. Raghuwanshi, and P. D. Shobhane, "An overview of similarity-based methods in predicting social network links: A comparative analysis," *IEEE Access*, vol. 12, pp. 120 913–120 934, 2024.

[39] S. Abinash, "Two novel graph theory based algorithms for salient object detection," in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, 2018, pp. 1–6.

[40] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015. [Online]. Available: https://arxiv.org/abs/1511.08458

[41] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," 2018. [Online]. Available: https://arxiv.org/abs/1706.02216